



## UMAP, INTELIGENCIA COLECTIVA EXTRAÍDA DE LAS REDES SOCIALES



**Eneko Astigarraga, Josu Azpillaga, Luis Fernández y Aitzol Naberan**



**Eneko Astigarraga**, MBA por la *Univ. del País Vasco (UPV/EHU)* y licenciado en ciencias políticas y sociología por la *Univ. de Deusto*, es consultor senior para diversas organizaciones y empresas. Ha participado en proyectos de i+d relacionados con las redes sociales e internet. Es profesor de prospectiva en la *Deusto Business School (Univ. de Deusto)* y gerente de *CodeSyntax*.

*CodeSyntax. Internet Solutions*  
Azitaingo Industrialdea 3K. - 20600 Eibar, Gipuzkoa (España)  
<http://www.codesyntax.com/es> - <http://twitter.com/eastigarraga>  
[eastigarraga@codesyntax.com](mailto:eastigarraga@codesyntax.com)



**Josu Azpillaga** es ingeniero industrial por la *Universidad de Deusto*. Colaboró como investigador en el *Centro Tecnológico Ikerlan*, así como en *MIK (Mondragon Innovation & Knowledge)*. Ha dirigido proyectos relacionados con tecnologías web 2.0. Actualmente es responsable de i+d+i en *CodeSyntax*.

*CodeSyntax. Internet Solutions*  
Azitaingo Industrialdea 3K. - 20600 Eibar, Gipuzkoa (España)  
<http://www.codesyntax.com/es> - <http://twitter.com/jazpillaga>



**Luis Fernández**, licenciado en ciencias de la información por la *Universidad del País Vasco (UPV/EHU)*, es autor de varias publicaciones en torno a internet, las redes sociales y la generación, búsqueda y organización de contenidos digitales. Ha dirigido proyectos sobre tecnologías de la lengua. Es responsable de proyectos y co-fundador en *CodeSyntax*.

*CodeSyntax. Internet Solutions*  
Azitaingo Industrialdea 3K. - 20600 Eibar, Gipuzkoa (España)  
<http://www.codesyntax.com/es> - <http://twitter.com/luistxo>



**Aitzol Naberan** es ingeniero informático por la *Universidad del País Vasco (UPV/EHU)*. Ha participado en el desarrollo de diferentes proyectos tanto de i+d como proyectos comerciales en el ámbito de las redes sociales, el comercio electrónico y los contenidos digitales. Actualmente es desarrollador senior en *CodeSyntax*.

*CodeSyntax. Internet Solutions*  
Azitaingo Industrialdea 3K. - 20600 Eibar, Gipuzkoa (España)  
<http://www.codesyntax.com/es> - <http://twitter.com/aitzol>

### Resumen

El proyecto *Umap* es un intento práctico de obtener inteligencia colectiva del flujo de las redes sociales. Se analizan, filtran y valoran los enlaces que los usuarios comparten, de manera que la inteligencia colectiva se convierte en información estructurada. Por ejemplo, la información, comentarios y enlaces que comparten los usuarios pertenecientes a una comunidad determinada -lingüística, social, temática, organizativa-, son analizados tanto en tiempo real, y mediante métodos de computación se logra obtener tendencias con periodicidad determinada (horaria, diaria, semanal...). También, mediante algoritmos y búsqueda de simplicidad la información se convierte en noticieros automatizados en los que al valor propio de las noticias se le añaden las opiniones y la relevancia que le otorgan los usuarios de las redes sociales. *Umap* abre el camino a futuras aplicaciones en la extracción de información e inteligencia colectiva de comunidades que compartan intereses sociales, políticos, económicos, comerciales, empresariales, productos, marcas, tecnologías, etc., a través de redes sociales. Las primeras aplicaciones del proyecto *Umap* desarrolladas en *CodeSyntax* son ya visibles en <http://www.umap.eu>

### Palabras clave

Inteligencia colectiva, Microblogging, *Twitter*, Redes sociales, *Umap*.

**Title: *Umap*, collective intelligence extracted from social networks**

Artículo recibido el 03-06-11  
Aceptación definitiva: 12-07-11

## Abstract

The *Umap* project is a practical attempt to obtain collective intelligence from the flow of social networks: links that users share are analyzed, filtered and rated, making it feasible to convert collective intelligence into structured information. For example, information, comments and links that are shared by users belonging to a certain community —e. g., linguistic, social, thematic, organizational— are analyzed in real time to obtain trends at specified intervals (e. g., hourly, daily, weekly...). By means of simplicity search algorithms, the information flow becomes an automated news bulletin with its own value but to which opinions and the relevance that users of social networks give them are added. *Umap* opens the way for future applications centred in the extraction of information and collective intelligence from communities that share social, political, economical, trading, business, products, brand, technological and other interests through social networks. The first applications of the *Umap* project developed by *CodeSyntax* are already available at <http://www.umap.eu>

## Keywords

Collective intelligence, Microblogging, *Twitter*, Social networks, *Umap*.

**Astigarraga, Eneko; Azpillaga, Josu; Fernández, Luis; Naberan, Aitzol.** “*Umap*, inteligencia colectiva extraída de las redes sociales”. *El profesional de la información*, 2011, septiembre-octubre, v. 20, n. 5, pp. 542-547.

<http://dx.doi.org/10.3145/epi.2011.sep.08>

## 1. Introducción

Las redes sociales —en general, no solamente las de internet— son estructuras sociales compuestas de grupos de personas, conectadas por uno o varios tipos de relaciones tales como amistad, parentesco, intereses comunes o para compartir conocimientos. También pueden definirse como sistemas abiertos y en construcción permanente, que involucran a conjuntos que se identifican con las mismas necesidades y problemáticas, y que se organizan para potenciar sus recursos.

Por otra parte, las redes son formas de interacción social, definida como un intercambio dinámico entre personas, grupos e instituciones en contextos más o menos complejos. En internet el concepto de comunidad es cada vez más importante, pues contextualiza y personaliza los contenidos que se producen en su seno. Para que los miembros de una determinada comunidad incrementen sus conocimientos de una forma óptima, los contenidos que reciben deben estar etiquetados, valorados y comentados de acuerdo con sus necesidades. Al tiempo cada usuario es productor de valor añadido a los contenidos, lo que luego permite personalizar la oferta de acuerdo con su perfil.

Para clasificar el contenido pueden usarse varias técnicas automáticas: reconocimiento de voz y de imagen, marcado semántico, posicionamiento geográfico, minería de datos... que a su vez permiten enriquecer el contenido con nuevas características que sirven para incrementar el valor de la información.

En la denominada sociedad red, termino profusamente difundido y descrito por **Manuel Castells**<sup>1</sup> cualquier nodo de la misma es tan capaz como cualquier otro de transmitir un mensaje y como señala el sociólogo **Paul Pierre Levy** “Nadie sabe todo. Todos sabemos algo. Todo el conocimiento reside en las redes”<sup>2</sup>. En definitiva, en la sociedad en red el conocimiento se genera de forma colectiva. Así, **George Pór** definió el fenómeno de la inteligencia colectiva como la capacidad de las comunidades humanas de evolucionar hacia una complejidad y armonía mayor, tanto por medio de mecanismos de innovación como de diferenciación e integración, competencia y colaboración<sup>3</sup>.

Redes sociales como *Twitter*, *Facebook*, *Tagzania* y otras muchas permiten generar y compartir información, generando lo que se ha venido en denominar inteligencia colectiva generada a partir de las redes sociales.

Previsiblemente, el resultado obtenible a través de otras redes sociales sería similar. Es decir, puede hablarse de una inteligencia generada a través del flujo e interacción de las redes sociales que tal y como proponen diferentes autores puede denominarse “inteligencia colectiva” y que en diferentes contextos tratamos de aflorar a través de la aplicación *Umap*.

**Bullmore**, en un artículo titulado “What does *Twitter* have to do with the human brain?”<sup>4</sup>, sugiere que aunque nos gusta pensar que el cerebro humano es especial, algo diferente a los procesos de generación y procesamiento de información, en el fondo todos los sistemas de información y procesamiento funcionan de manera similar. **Bullmore** ha realizado varios experimentos (usando el tag *#csftwitterbrain*) y afirma que la red *Twitter* se puede comparar al cerebro humano.

“La red de *Twitter* se puede comparar al cerebro humano”

Aunque la experiencia acumulada con *Umap* no nos permite afirmar tanto, las primeras aplicaciones llevadas a cabo para extraer inteligencia colectiva generada en comunidades lingüísticas minoritarias —*Umap euskara*, *Umap català* y *Umap Cymraeg* (gaélico), todas ellas accesibles desde <http://www.umap.eu>, nos han permitido comprobar algunas de las afirmaciones anteriores.

## 2. Metodología: extracción de información desde las redes sociales de microblogging

Como se señala en la introducción, extraer información relevante de los inputs generados por los distintos usuarios de las redes sociales requiere definir, en un primer momento, la

Register for free at <https://www.scipedia.com> to download the version without the watermark

comunidad o colectivo a estudiar, que estará caracterizado por compartir algún interés o rasgo. Éste puede ser el uso de un determinado idioma (catalán, gaélico, euskara...), tener un mismo interés (coches, motos, perfumes...), ser usuario de una marca o empresa, algo social o político, pertenecer a una comunidad geográfica...

En el caso de la aplicación práctica de *Umap* que se presenta aquí el rasgo común al colectivo es el idioma, y la aplicación analizada es *Twitter*, cuyo flujo de información es recogido, filtrado y tratado.

Es decir, se trata de leer, interpretar y extraer información relevante de todas las personas que hablan -en este caso escriben- en un determinado idioma (euskera, catalán, gaélico), y analizar únicamente los tweets que cumplan con el requisito de estar escritos en uno de esos idiomas.

El proceso seguido por *Umap* es:

- Recoger en tiempo real todos los inputs (tweets) de todos los usuarios de la comunidad lingüística.
- Detectar posibles nuevos usuarios de esa comunidad (nuevos usuarios de ese idioma).
- Discriminar los mensajes de los usuarios según un parámetro concreto. En este caso no todos los tweets de los usuarios están escritos en dicho idioma. Sólo nos interesan algunos de ellos.
- Analizar los tweets y extraer trends o palabras clave, así como establecer un ranking de links compartidos por esa comunidad.

A continuación pasamos a exponer una aplicación concreta de *Umap* aplicada a comunidades lingüísticas. Como hemos indicado, este tipo de análisis se puede aplicar a otras comunidades que compartan intereses sociales, políticos, comerciales, empresariales...

### 3. *Umap* como marcador de comunidad lingüística

*Umap* es un sistema de extracción de datos a partir de micromensajes (tweets) que obtiene como resultado información nueva.

En una primera fase se usa el *streaming api* de *Twitter*. *Twitter* ya extrae información de subconjuntos de los mensajes que pasan por sus servidores y también construye información nueva sobre ellos (básicamente, los *trending topics*), pero su lógica y su alcance son distintos a los nuestros.

Los *trending topics* o palabras claves que usa *Twitter* se basan en criterios geográficos, no lingüísticos. De momento esa función está limitada a ciertos países y ciudades. Además, la información agregada obtenida no pasa filtros ni lingüísticos ni geográficos. Es decir, la lista de *trending topics* tiene significación y relevancia local, pero no la información que se obtiene clicando en ellos, que sólo lo es con respecto al criterio de filtro.

Por ejemplo, si en España un *hashtag* como *#PerezReverteFacts* pasa a ser trending topic cabe pensar que clicando ahí, la información acumulada que se obtiene es de tweets españoles, ya que Arturo Pérez-Reverte es un autor y personaje público español; no obstante, si el topic español del

momento es *#tsunami* la información que se obtiene en ese enlace es la de todos los mensajes que llevan ese identificativo sean de España o de otros lugares, estén en español, sueco o japonés.

En el caso que presentamos, el objetivo de *Umap* era precisamente hacer que el idioma fuera un factor definidor de comunidad o de la información, y convertir eso en el eje sobre el que construir información adicional. Por ello el primer paso del proceso fue investigar el modo en que se podían identificar los mensajes en un idioma dado en *Twitter*.

Nota: Los *hashtags* son palabras precedidas del signo almohadilla, que los usuarios crean para etiquetar mensajes, y así poderlos recuperar en otro momento, por ejemplo *#Biblioteca* o *#NoAlCanon*.

#### 3.1. Detección de idioma

Los metadatos de los que provee el api de *Twitter* sí identifican un componente idiomático, pero es un sistema bastante deficiente, por dos motivos:

- Sólo identifica un número finito de idiomas.
- Asigna un idioma a todos los tweets.

A consecuencia de ello, por ejemplo, los tweets en catalán quedan identificados erróneamente siempre: no es uno de los idiomas en los que funciona su sistema de detección, con lo que no sólo falla para el caso catalán, sino que los tweets en catalán contaminan los resultados en español, francés, italiano... Esto ocurre porque a los mensajes en catalán de *Twitter* el sistema de detección siempre les asigna otro idioma, normalmente una lengua romance cercana.

A la vista de que los datos del api de *Twitter* eran insuficientes, quedó claro que la detección de idioma debíamos hacerla con nuestra aplicación.

En las primeras pruebas que hicimos usamos el api de detección de idioma de *Google*. No acierta siempre, pero en nuestra opinión su fiabilidad parece muy elevada, y nos sirvió para comprobar que el resto de los componentes de *Umap* podían funcionar. Sin embargo, usar el api de *Google* para filtrar miles de mensajes individuales en un proceso continuo es inviable: no lo permiten las condiciones de uso del api, que restringen el número de peticiones por hora.

Por lo tanto, el siguiente paso fue encontrar detectores de idiomas que funcionaran en nuestros propios servidores. Después de realizar pruebas con varios de ellos se optó por una solución modular: utilizar uno u otro dependiendo del idioma.

Hemos hecho pruebas con varios detectores: *Langid* (Perl), *Textact* (Perl) y *Guess Language* (Python).

La modularidad también nos permite añadir niveles extra de programación. Esto se ha revelado necesario ya que los mensajes de *Twitter* son a lo sumo de 140 caracteres, y la detección de idioma en un mensaje tan corto no es lo mismo que en un texto más largo (una página web, por ejemplo).

Una de las particularidades de varios detectores de idiomas es que dan como respuesta más de una lengua: en documentos de cierta longitud es muy útil puesto que así se detecta la presencia de más de un idioma, pero en micromen-



sajes de *Twitter* la multiplicidad de idiomas la mayoría de las veces suele ser signo de error, sobre todo si el mensaje apenas tiene 30 ó 40 caracteres. Además, pese a ser cortos los mensajes de *Twitter* pueden también ser bilingües e incluir jerga tecnológica en inglés o títulos de obras (películas, libros) en un idioma que no es el usado en la conversación.

Pese a ser cortos, los tweets pueden también ser bilingües e incluir jerga o títulos de obras en un idioma distinto del usado en la conversación

En el caso del detector de euskera hemos implantado una operación extra para hacer frente a este problema: sólo considerar que se trata de euskera si el número de caracteres es >25.

En el caso del catalán el módulo de lengua es aún un poco más complicado puesto que los detectores pueden dar como catalán textos escritos en otros romances: particularmente, hemos comprobado que ello ocurre con mensajes escritos en español de varios países latinoamericanos, sobre todo si son cortos y los usuarios prescinden de acentos o de la corrección gramatical. El módulo de detección de idioma se enriqueció con un filtro previo de *stopwords* positivos y negativos: los primeros sirven para identificar el mensaje como catalán y los segundos para identificarlo como castellano.

### 3.2. Detección de usuarios en un idioma

Antes de detectar el idioma hay que detectar los usuarios. *Umap* no filtra todos los mensajes que generan los 200 millones de usuarios de *Twitter*: primero se detectan los usuarios susceptibles de usar un idioma dado, y luego se filtran sus mensajes.

Cabe suponer que todos los usuarios de los 3 idiomas en los que se ha trabajado son bilingües cuanto menos. Hay cuentas de *Twitter* perfectamente monolingües en los tres idiomas, sobre todo cuentas robóticas asociadas a medios de comunicación que se alimentan automáticamente vía RSS o algún otro método, pero en el caso de usuarios humanos, aun cuando pueden ser también monolingües en su comportamiento, potencialmente siempre pueden tuitear en otro idioma.

La mecánica es, pues, detectar usuarios que tuitean en un idioma dado y luego filtrar todos sus mensajes, para procesar aquellos que están en el idioma deseado, desechando el resto.

El método de detección de usuarios no sigue ninguna técnica complicada: partiendo de un número reducido de usuarios que marcamos como hablantes de la lengua por haberlo comprobado directamente (humanamente,

claro), *Umap* empieza a seguirlos y detecta más usuarios de manera simple: es también hablante aquel a quien un hablante interpela directamente.

Los mensajes de *Twitter* empiezan con un nombre del usuario, tal que @fulano.

Por ejemplo:

**@asarasua** lehengoan Antzuolan inauguratu zuten duela urtebetetik zabalik dagoen bidegorria... eta oraindik osoa bukatzeke dago!

Si un usuario interpela a @asarasua en euskera se puede asumir que @asarasua puede entender y usar este idioma. El sistema sólo se aplica con nombres de usuario en posición inicial, puesto que son habituales menciones a otros usuarios en posición intermedia:

**asko gustatzen zait @stephenfry** aktore gisa, eta bere iritziz intelektualak ere bai

Si el usuario quisiera interpelar directamente a Stephen Fry, actor y celebridad británica, es probable que lo haga directamente en inglés y empezando:

**@stephenfry** I really did like your last article at The Guardian

No obstante, aunque no es el modo habitual en que tuitea la gente, si es posible que por la forma en que se construye la frase un usuario empiece un tweet en euskera con...

**@stephenfry** benetan atsegin dudan aktore bat da (Stephen Fry es un actor que me encanta de verdad)

Tras captar el tweet anterior, *Umap* añadiría a Stephen Fry a su lista de usuarios en euskera y todos sus mensajes empezarían a ser filtrados. No obstante probablemente ninguno será marcado como en euskera, y por lo tanto, ni contaminará el contenido ni añadirá más falsos usuarios al listado. Como provisión de seguridad además *Umap* tiene un mecanismo para limpiar usuarios: periódicamente se revisa el listado de usuarios y se borran aquellos que nunca tuitean en el idioma objeto de estudio. Es así como Stephen Fry desaparecerá de la base de datos.



Figura 1. Intercambio de información vía microblogging de usuarios en gaélico, recogida en *Umap Cymraeg*. En la columna de la derecha *trending topics* de las últimas 2 horas, las últimas 24 horas y la última semana.



Figura 2. Archivo de temas en catalán

### 3.3. Información agregada

Una vez definida la comunidad de la que queremos extraer información y el subconjunto de mensajes que queremos procesar (ambos parámetros marcados por el idioma), *Umap* extrae dos conjuntos de información principales:

- Trends o palabras clave
- Noticias o enlaces

### 3.4. Trends

Detectar los trends o palabras clave no es un proceso complejo en sí mismo: se detectan las palabras en los tweets, y se hace un conteo con las acumuladas en los diferentes plazos de tiempo: últimas 2 horas, últimas 24 horas, última semana o último mes (30 días).

El recuento en el caso de dos horas es una variable que modulamos según el volumen de tweets: en principio llamamos a ese primer plazo “ahora”, que puede necesitar más o menos tiempo para ser relevante. Se sigue un proceso bastante simple: limpieza, valoración de importancia del tweet, etc.

La limpieza consiste en:

- detectar y eliminar las palabras vacías (las más comunes del idioma);
- eliminar urls y nombres de @usuario;
- eliminar signos de puntuación.

Para determinar la importancia del tweet se considera que si menciona a otros @usuarios y tiene enlaces, es más rico y los términos incluidos en el mismo merecen una ponderación positiva. Un argumento de ponderación que aún no hemos implementado es la determinación de usuarios robóticos; en este caso, el factor de ponderación será negativo, puesto que creemos que los tweets humanos tienen más valor que los que puedan generar servicios de noticias automatizados.

Finalmente se aplica un factor de ponderación a los términos capturados, distinguiendo las palabras que son hashtag: si una palabra normal tienen el valor 1, los hashtags tienen un valor N. En esta etapa se puede hacer alguna distinción más, como dar un valor intermedio entre hashtags y palabras comunes a los nombres propios (personas, lugares, marcas y productos).

### 3.5. Enlaces

El otro tipo de información que genera *Umap* es el recuento de enlaces contenidos en los tweets. Por un lado se de-

termina qué enlaces están siendo más compartidos en las últimas horas, con lo que se puede llegar a conocer qué es noticia, o qué es lo que la comunidad considera digno de ser compartido. Por otro lado, los tweets en los que aparecen enlaces se agrupan para ver qué comenta la gente sobre ellos.

En este proceso no se tiene en cuenta el idioma en que está el contenido al otro lado del enlace. Podría desarrollarse una versión tal, pero creemos que hay más valor en que el comentario de los usuarios se ha hecho en el idioma

comunitario, independientemente del contenido referido. Es, por ejemplo, muy usual que un vídeo impactante un día dado se comente en multitud de idiomas.

El conteo tiene varias fases:

- determinar los enlaces;
- parsear o analizar su contenido para extraer los datos básicos;
- ordenar por relevancia, creando un noticiero;
- archivar, y previsión de reaparición del enlace.

La determinación de enlaces es compatible con los distintos sistemas de acortado de urls: *bit.ly*, *goo.gl* o el mismo *t.co* de *Twitter* resuelven urls de modo ágil y sólo se contabilizan las urls finales, lógicamente.

El analizador de *Umap* se activa a partir de que un enlace obtiene dos menciones. Entonces un robot revisa el contenido del enlace (una página web, por ejemplo), y si robots.txt lo permite extrae título, algunas líneas de texto, un thumbnail de lo que se estima sea la imagen más relevante presente en la página, etc.

El sistema tiene prevista una herramienta modular para que cierto tipo de urls puedan ser procesadas de manera particular. Por ejemplo, si el sitio *Foo.com* es relevante para un idioma dado, y el sistema estándar de parseo o análisis no obtiene información del mismo debido al html peculiar que tiene, se puede crear un método de parseo específico para *Foo.com*.

Igualmente, el parseador está adaptado al estándar *Oembed* que usan diversos servicios de internet, que proveen a los robots con un contenido concreto que se puede embeber o copiar fácilmente en otros sistemas. Por ejemplo, *YouTube* es un sitio que usa *Oembed*, y por ello el robot de *Umap* lo tiene más fácil para poder embeber el reproductor del vídeo y otros metadatos de su contenido.

Los enlaces parseados pasan a una base de datos donde periódicamente se hace un conteo de su relevancia. En este caso el factor ponderador es el tiempo, y está definido por tramos por horas. Si la mención de un enlace vale 1, el que la mención sea en las últimas 2 horas tiene un factor de multiplicación x4, si la mención es entre 2 y 5 horas, x3, más vieja aún x2... Los factores y los tramos temporales son modulares y editables, pero su función básica es que si recientemente un enlace obtiene varias menciones, llegue al puesto principal de noticias relevantes, aunque en las últi-

mas 48 horas tenga menos menciones que una noticia que generó varios enlaces en el día de ayer.

En principio, hay un máximo de tiempo marcado para que un enlace entre en el conteo de relevancia: 48 h. Este tiempo se puede variar, pero no parece lógico prolongarlo más allá de dos días, pues en “tiempo de internet” pasado este lapso parece que todo deja de ser noticia. Tras ese tiempo, el enlace queda archivado, con los tweets en los que se mencionó asociados a él.

Puede suceder que un enlace concreto, caducado tras dos días de comentarios, pueda volver a ser actualidad. Por ejemplo, una url de la empresa *Foo.com* que fue comentada hace 5 semanas por una cuestión determinada, hoy está otra vez de actualidad por otro tema. En este caso *Umap* renueva el proceso de parseo (puesto que el contenido de portada de *Foo.com* puede haber variado) y también guarda los nuevos tweets asociados a esta nueva fase, prescindiendo de los generados hace 5 semanas. No obstante, se guardan todas las referencias, y aunque de momento no hemos desarrollado interfaces sobre esta opción, cabe imaginar un histórico de menciones y comentarios de una url dada, de modo que se pueda mostrar el historial de *Foo.com*: cuándo fue noticia tal url (en distintos momentos de su historia), qué contenido mostraba cuando fue noticia, y qué comentarios generó en cada uno de esos momentos de “fama”.

En la escala de *tiempo de internet* parece que todo deja de ser noticia transcurridas 48 h

#### 4. Conclusiones y líneas futuras de investigación

En los párrafos que preceden hemos expuesto un caso de uso de *Umap* aplicado a comunidades lingüísticas minoritarias. Los resultados obtenidos son alentadores, el sistema es capaz de obtener información relevante para y referente a las tres comunidades seleccionadas: euskara, catalana, gaélica. El método se puede extrapolar a otros intereses compartidos por una comunidad: comerciales, marcas, productos o tecnologías, sociales, económicos, políticos, etc.

**Bilduren zerrendak inpugnatzea aginduko du gaur Espainiako Gobernuak - EITB Albisteak Politika**  
 EITB Albisteak - eitbcomBerriak | Api. 20, 2011, 10:30 a.m. | eitb.com  
 Bilduren zerrendak inpugnatzea aginduko du gaur Espainiako Gobernuak <http://bit.ly/g4x61E> +2  
 Francisco Caamaño Justizia ministroak proposatuta, gaurko Ministroen Kontseiluak baimena emango dio Estatuko Abokatutzari Bildu koalizioaren zerrenden aurka egin ahal izateko, baldin eta segurtasun indarrek aurkeztutako txoztenek Batasunak

**ARGIA.com - Udal eta foru hauteskundeetako zerrendak**  
 Argia Astekaria - argia | Api. 20, 2011, 11:17 a.m. | argia.com  
 Udal eta foru hauteskundeetako zerrendak <http://www.argia.com/albiste/udal-eta-foru-hauteskundeetako-zerrendak> +1

**HitzaPasa » Blog Archive » HitzaPasa “Lan Munduko Euskara Planak” 1-1 - Jokatu eta ikasi**  
 Maite Goñi - euskaljakintza | Api. 19, 2011, 11:59 a.m. | euskaljakintza.com  
 Emuneko langileek lan munduko euskara planak buruzko hitzapasa bidali didate. Ez al da sekulakoa? Mila esker! :) <http://bit.ly/tZtQZ6> +5

Figura 3. Noticiero automatizado en *Umap* euskera.

El sistema aplicado por *CodeSyntax* en *Umap.eu*<sup>5</sup> permite extraer, almacenar y tratar de forma automatizada y en función de diferentes parámetros los temas de interés para esas comunidades y las redes sociales que generan. Se han obtenido noticieros automatizados que seleccionan y ponderan las informaciones más relevantes, ofreciendo un resumen de la inteligencia colectiva extraída de tales redes sociales. Se ha validado la utilidad de la herramienta y se han abierto posibilidades de aplicación en otros campos como los arriba mencionados. Su aplicación en otro tipo de redes sociales además del microblogging constituye otra de las líneas de investigación futura.

#### 5. Referencias bibliográficas

1. Castells, Manuel. *La sociedad en red*. Madrid: Alianza Editorial, 2005.
2. [http://en.wikipedia.org/wiki/Social\\_network](http://en.wikipedia.org/wiki/Social_network)
3. [http://en.wikipedia.org/wiki/Collective\\_intelligence](http://en.wikipedia.org/wiki/Collective_intelligence)
4. Bullmore, Ed. *Cambridge neuroscience*, March 2011 <http://www.admin.cam.ac.uk/news/dp/2011031105>
5. <http://www.umap.eu>
6. <http://www.codesyntax.com/taller-cs/blog/topics/umap>